


 COURSE DEVELOPMENT PROGRAM +
 financed by
 Austrian
 Development Cooperation

Programiranje za Internet



predavanja
 v.as.mr. Samir Lemeš
 slemes@mf.unze.ba

29. Baze podataka

- API funkcije za pristup bazama podataka
- Baze podataka na web sajtovima
- Windows vs. LINUX vs. ASP vs. PHP
- Spajanje na bazu podataka
- Manipulacija podacima

API funkcije

- Da bi aplikacija komunicirala sa bazom podataka, potrebno je određeno "sredstvo". To sredstvo su API funkcije za pristup bazama podataka.
- Sama baza podataka nije ništa više nego "skladište" koje sadrži odgovarajuću reprezentaciju, odnosno apstrakciju procesa iz stvarnog života.
- Da bi podaci koji se čuvaju u bazi postali informacija od ikakvog značaja, potrebna je aplikacija koja zna kako da transformiše te podatke u oblik koji krajnji korisnik želi da vidi.
- Sredstvo pomoću kojeg će aplikacija "pričati" sa bazom podataka se naziva database API.

API funkcije

- U hronološkom redu njihovog pojavljivanja najbitniji API-ji su:
 - Open Database Connectivity (ODBC),
 - Object Linking and Embedding Database (OLE DB),
 - Java Database Connectivity (JDBC) i
 - ActiveX Data Objects (ADO).
- Kreatori svih ovih API-ja su Microsoft i JavaSoft. Microsoft još koristi i termin Universal Data Access (UDA) kao krovni termin koji označava ODBC, OLE DB, ADO i Remote Data Service (RDS).

API funkcije

- **ODBC** kao najstariji de jure standard je ujedno i najrašireniji i Microsoft i JavaSoft su prilikom pravljenja novijih standarda vodili računa o njihovoj kompatibilnosti sa ODBC.
- Novi API-ji imaju objekte ili komponente koje putem ODBC adaptera pristupaju SQL bazama podataka.
- Microsoft strategija je budući pomak ka OLE DB i ADO tehnologijama, dok JavaSoft želi pomak ka Javi i JDBC standardima.

API funkcije

- Aplikacije za upravljanje bazama podataka pozivaju funkcije u ODBC interfejsu, a ODBC putem svojih drajvera za baze podataka aplikaciji vraća podatke iz baze. Ovi drajveri su analogni drajverima za štampače.
- Kada npr. program za obradu teksta nešto štampa, sve komande za štampanje se upućuju operativnom sistemu, koji upotrebljava odgovarajući drajver za željeni štampač.
- Isto tako, kada aplikacija za obradu podataka želi da pristupi bazi, ona ODBC-u upućuje zahtjev za podacima, a ODBC koristi odgovarajući drajver za pristup toj bazi podataka.

API funkcije

- **OLE DB** je u neku ruku nova verzija ODBC.
- OLE DB nadograđuje ODBC tako što, na jednako uniforman način na koji to radi ODBC sa relacionim bazama, omogućava pristup heterogenim izvorima podataka i vrši njihovu prividnu homogenizaciju, pojednostavljujući tako rad sa podacima.
- Microsoft strateški vodi OLE DB kao novu tehnologiju za pristup svim vrstama podataka i kao takav će u budućnosti u potpunosti potisnuti ODBC.

API funkcije

- **JDBC** je ekvivalent ODBC tehnologije namijenjen upotrebi prilikom razvoja aplikacija u Java programskom jeziku.
- Koristi sličnu arhitekturu zasnovanu na API pozivima i drajverima za razne vrste sistema za upravljanje bazama podataka (DBMS).
- Sun (kreator Java platforme) je u JDBC ubacio i takozvani JDBC-ODBC "most" - specijalni drajver koji aplikaciji koja koristi JDBC omogućava pristup ODBC izvorima podataka.
- Kako je većina baza podataka u trenutku pojave JDBC već imala gotove ODBC drajvere, ovo je značilo da Java aplikacije mogu pristupati svim takvim bazama, iako JDBC drajveri za njih nisu (bili) dostupni.

API funkcije

- Prvi objektni model je **DAO** (Data Access Objects), koji je sastavni dio VisualBasica.
- Originalno zamišljen kao set objekata za pristup MS Access bazama podataka preko MS Jet sistema za upravljanje podacima (Jet je prvi Microsoft DBMS na kojem je zasnovan Access), DAO također omogućava pristup ODBC izvorima podataka.
- DAO je lakši za korištenje od standardnih ODBC API funkcija, ali ne pruža isti nivo kontrole kao ODBC APL.

API funkcije

- **ADO** je nova tehnologija iz Microsofta čiji je cilj da zamijeni DAO kao standardni objektni model za pristup bazama podataka.
- Programiranje pomoću ADO modela ne pruža toliko kontrole kao direktan rad sa OLE DB pozitivima, ali je ADO dizajniran tako da je broj slojeva između aplikacija i baze podataka sveden na apsolutni minimum, što mu daje znatno bolje performanse.
- Prednost ADO modela je zasnovanost na ActiveX tehnologiji, što ga čini upotrebljivim u svim jezicima koji podržavaju tehnologiju, uključujući i skriptne jezike poput VBScripta i JavaScripta, što OLE DB ne može postići.

Baze podataka na web sajtovima

- Danas su najčešće dvije tehnologije koje se koriste za razvoj dinamičkih web sajtova:
 - ASP / MS SQL
 - PHP / MySQL
- Osnovna razlika je u platformi servera (Microsoft server, odnosno Linux server)

Windows vs. LINUX ASP vs. PHP

- Iako se za razvoj web aplikacija mogu koristiti mnogi skriptni i programski jezici (Perl, Python, C), većina programera odlučuje se na PHP ili ASP.
- ASP će kvalitetno raditi jedino na Windows platformi, iako postoje Apache moduli koji ASP-u donose podršku i za Unix.
- ASP stranice su najčešće napisane u VBScriptu, što je uobičajeni, ali ne i jedini način pisanja ASP aplikacija.
- Sa druge strane, PHP postoji i za Windows, gdje je donekle lišen funkcionalnosti (neke funkcije, doduše rijetke, raditi će samo na Unix/Linux verziji Apache web servera).

Windows vs. LINUX ASP vs. PHP

- Za učenje PHP-a, Apache za Windowse ili PHP modul za PWS (Personal Web Server) ili IIS (Internet Information Server) poslužit će sasvim dobro.
- Programeri sa iskustvom sa Visual Basicom, ASP doslovno neće morati učiti.
- Programeri sa iskustvom u C-u ili Perlu, lakše će savladati PHP.

Spajanje na bazu podataka

- DSN (data Source Name) metoda spajanja na bazu u ASP:

```
Set baza = Server.CreateObject  
("ADODB.Connection")  
Baza.Open "DSN=naziv_dsn; UID=username;  
PWD=password"
```

Spajanje na bazu podataka

- DSN-less metoda spajanja na Accessovu bazu podataka u ASP-u:

```
Set baza = Server.CreateObject  
("ADODB.Connection")  
Baza.Open "DRIVER={Microsoft Access Driver  
(*.*.mdb) }; DBQ=" & Server.MapPath  
("naziv_baze.mdb") & "; UID=username;  
PWD=password"
```

Spajanje na bazu podataka

- DSN-less metoda spajanja na Microsoft SQL bazu podataka u ASP-u:

```
Set baza = Server.CreateObject  
("ADODB.Connection")  
Baza.Open "DRIVER={SQL Server};  
Server=ime_servera; Database=ime_baze;  
UID=username; PWD=password";
```

Spajanje na bazu podataka

- OLEDB metoda spajanja na Access bazu podataka u ASP-u:

```
Set baza = Server.CreateObject  
("ADODB.Connection")  
Baza.Open "Provider=Microsoft.Jet.OLEDB.4.0;  
Data Source=" & Server.MapPath  
("naziv_baze.mdb") & "; User ID=admin;  
Password=;"
```

Spajanje na bazu podataka

- OLEDB metoda spajanja na Microsoft SQL bazu podataka u ASP-u:

```
Set baza = Server.CreateObject  
("ADODB.Connection")  
Baza.Open "Provider=SQLOLEDB; Data  
Source=ime_servera; Initial  
Catalog=ime_baze; User ID=username;  
Password=password;"
```

Spajanje na bazu podataka

- DSN metoda spajanja na bazu podataka u PHP-u:

```
$baza = odbc_connect ( 'SYSTEM_DSN' ,  
    'username' , 'password' );
```

Spajanje na bazu podataka

- DSN-less metoda spajanja na Access bazu podataka iz PHP-a (Windows):

```
$baza = new COM ("ADODB.Connection");  
$konekcija = "DRIVER= {Microsoft Access  
    Driver (*.mdb) } ; ";  
$konekcija .=  
    "DBQ=baza_podataka.mdb;uid=username:p  
    wd=password;";  
$baza->open ($konekcija);
```

Spajanje na bazu podataka

- DSN-less metoda spajanja na MySQL bazu preko ASP-a:

```
Konekcija = "driver={MySQL}; server=host;  
    uid=username:  
    pwd=password;database=baza"  
Set baza = Server.CreateObject  
    ("ADODB.Connection")  
Baza.open Konekcija
```

Spajanje na bazu podataka

- Spajanje na MySQL bazu iz PHP-a (Unix):

```
$link = mysql_connect ("host", "username",  
    "password")  
    Or die ("Neuspjesno spajanje na server");  
Mysql_select_db ("baza")
```